SPECIFICATION AMENDMENT

Please change title of the invention, on page 1, line 13, to the following title:

Space Allocation in a Write Anywhere File System

Please add the following new heading and paragraph at page 1, line 14:

Cross-Reference to Related Application

This application hereby incorporates by reference and claims benefit of U.S. Application No. 09/642,065, filed 8/18/2000, now U.S. Patent No. 6,636,879.

Please replace the paragraph at page 6, lines 8-10, with the following amended paragraph:

- U.S. Patent Application Serial No. 09/642,063, Express Mail Mailing No. EL 524781089US, filed August 18, 2000, in the name of Blake LEWIS, attorney docket number 103.1033.01, now U.S. Patent No. 6,640,233, titled "Reserving File System Blocks"

Please replace the paragraph at page 12, line 18, to page 13, line 7, with the following amended paragraph:

The volume 101 is broken up into a collection of allocation areas 102. Each allocation area is composed of a set of stripes with consecutive DBN values, such as 4,096 stripes. The allocation areas 102 can be broken down further into a group of ranges per disk 104, 106, 108 and 110. In a preferred embodiment, the ranges 104, 106, 108 and 110 include a group of 1,024 blocks that are described by the spacemap's 16 bit binary number for allocation usage. A binary number quantifying the amount of used (unavailable for writing) blocks represents the allocation areas 102 in the spacemap. A low value of the binary number for a particular allocation area 102 represents a high number of blocks being available for being written to. Conversely, a high value represents a small number of blocks being available for allocation. The spacemap binary values are organized by VBN number. Therefore, the relationship between a used block count of an allocation area 102 ~~202~~ and the spacemap binary values is approximate.

Please replace the paragraph at page 13, line 17, to page 14, line 2, with the following amended paragraph:

The active map 226 of the active file system 201 ~~that~~ is a bitmap associated with the vacancy of blocks for the active file system 201. The respective snapmaps 254, 256, 258 and 260 are active maps that can be associated with particular snapshots 214, 216, 218 and ~~220 and an inclusive OR summary~~ 220. Summary map 224 is an inclusive OR of the snapmaps 254, 256, 258 and 260. Also shown are other blocks ~~226~~ 228 including double indirect blocks 230 and 232, indirect blocks 234,

236 and 238 and data blocks 240, 242, 244 and 246. Finally, Figure 2 shows the spacemap 280 including a collection of spacemap blocks of binary numbers 282, 284, 286, 288 and 290.

Please replace the paragraph at page 15, lines 8-14 with the following amended paragraph:

Inode block 208 in the inode file 202 points to a set of blocks (1, 2, 3, ..., P) called the active map 226. Each block in the active map 226 is a bitmap where each bit corresponds to a block in the entire volume. A ~~"1"~~ "1" in a particular position in the bitmap correlates with a particular allocated block in the active file system 201. Conversely, a ~~"0"~~ "0" correlates to the particular block being free for allocation in the active file system 201. Each block in the active map 226 can describe up to 32K blocks or 128 MB. For a 6 TB volume, only 24 blocks are needed in the active map 226.

Please replace the paragraph at page 18, lines 9-21 with the following amended paragraph:

In a preferred embodiment, the large spacemap array binary number 284 (0000001111111101=1,021 in decimal units) tells the file system that the corresponding range is relatively full. In such embodiments, the largest binary number 00010000000000 (1,024 in ~~decimal~~ decimal) represents a range containing only occupied blocks. The smallest binary number 00000000000000 (0 in decimal) represents a range containing entirely free blocks. The small binary number 288 (0000000000001101=13 in decimal units) instructs the file system that the related range is relatively empty. The spacemap 280 is thus a representation in a very compact form of the

allocation of all the blocks in the volume broken into 1,024 block sections. Each 16 bit number in the array of the spacemap 280 corresponds to the allocations of blocks in the range containing 1,024 blocks or about 8 MB. Each spacemap block 280 has about 2,000 binary numbers in the array and they describe the allocation status for 16 GB. Unlike the summary map 214, the spacemap block 280 needs to be determined whenever a file needs to be written.